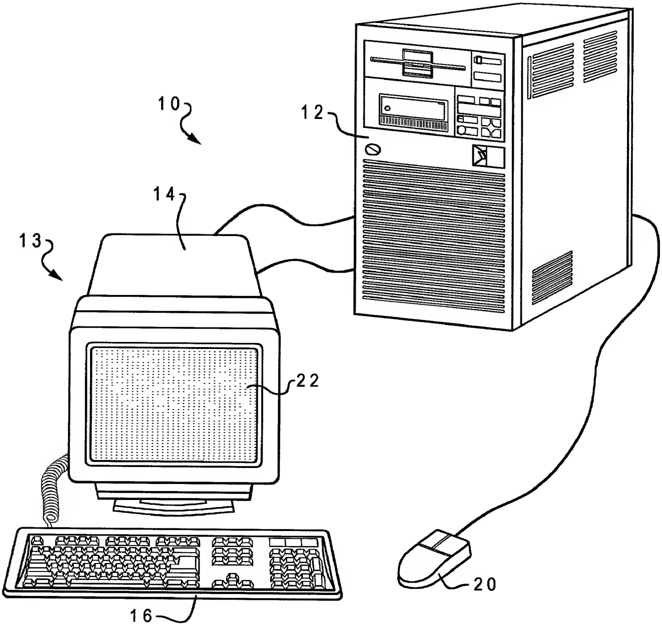


Fig. 1

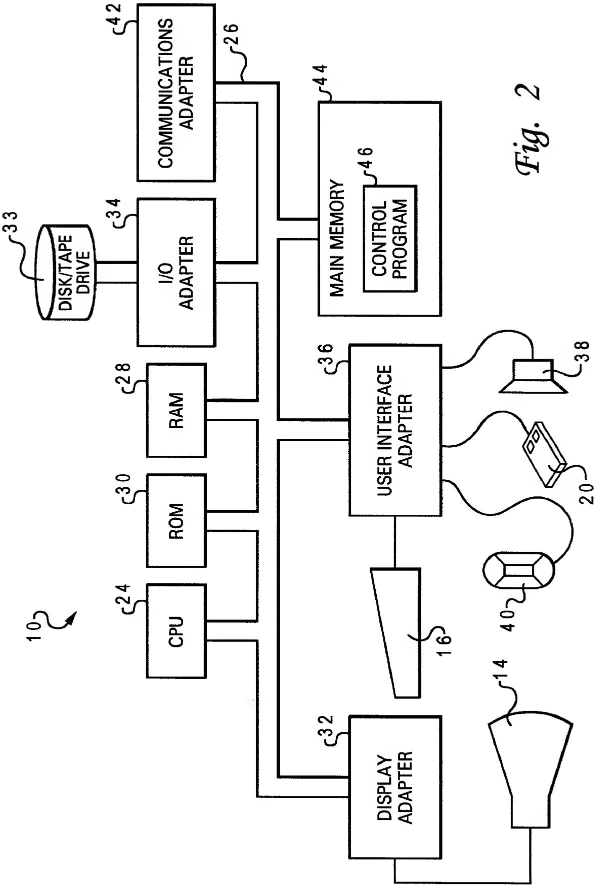

10
CPU
24
ROM
30
RAM
28
I/O ADAPTER
34
DISK/TAPE DRIVE
33
COMMUNICATIONS ADAPTER
42
26
MAIN MEMORY
44
CONTROL PROGRAM
46
USER INTERFACE ADAPTER
36
DISPLAY ADAPTER
32
14
16
40
20
38

Fig. 2

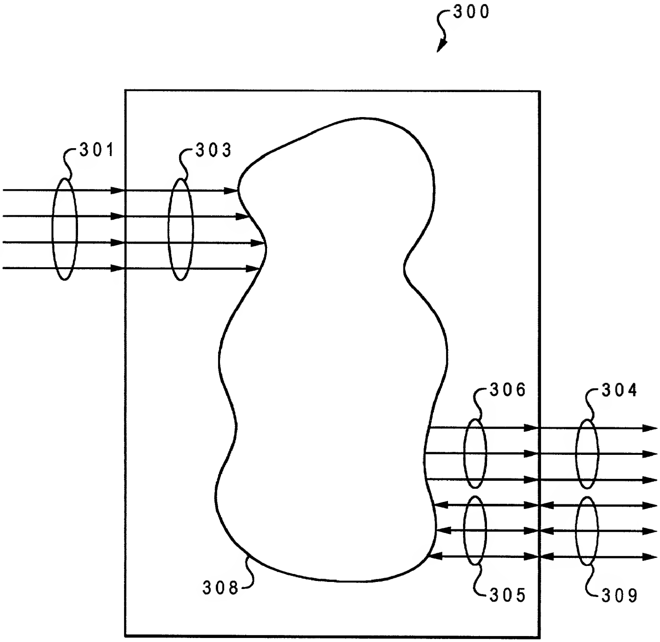


Fig. 3A

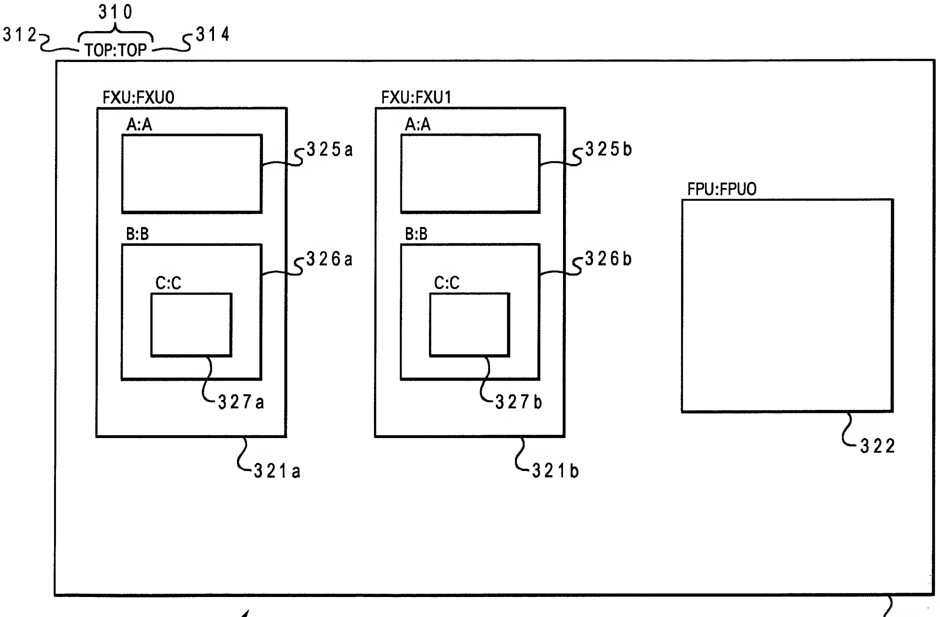

310
312
TOP:TOP
314
FXU:FXU0
A:A
325a
B:B
326a
C:C
327a
321a
FXU:FXU1
A:A
325b
B:B
326b
C:C
327b
321b
FPU:FPU0
322
320
329

Fig. 3B

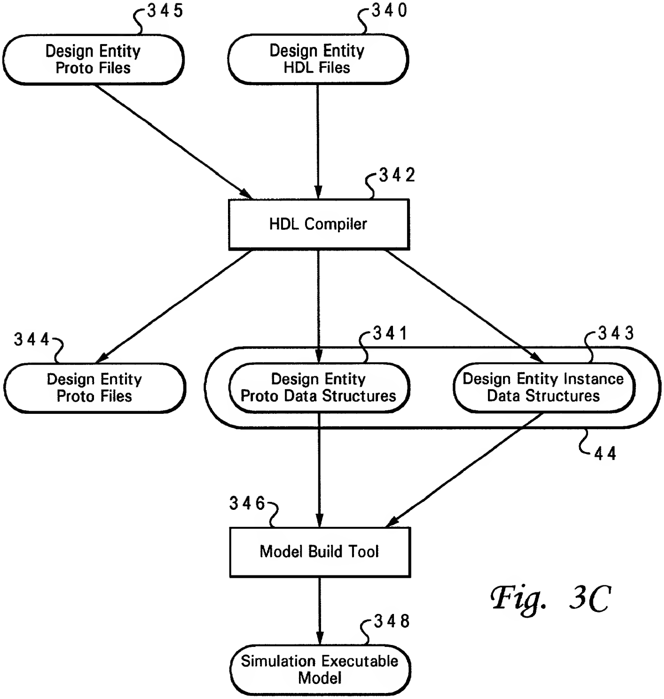

345
Design Entity Proto Files
340
Design Entity HDL Files
342
HDL Compiler
344
Design Entity Proto Files
341
Design Entity Proto Data Structures
343
Design Entity Instance Data Structures
44
346
Model Build Tool
348
Simulation Executable Model

Fig. 3C

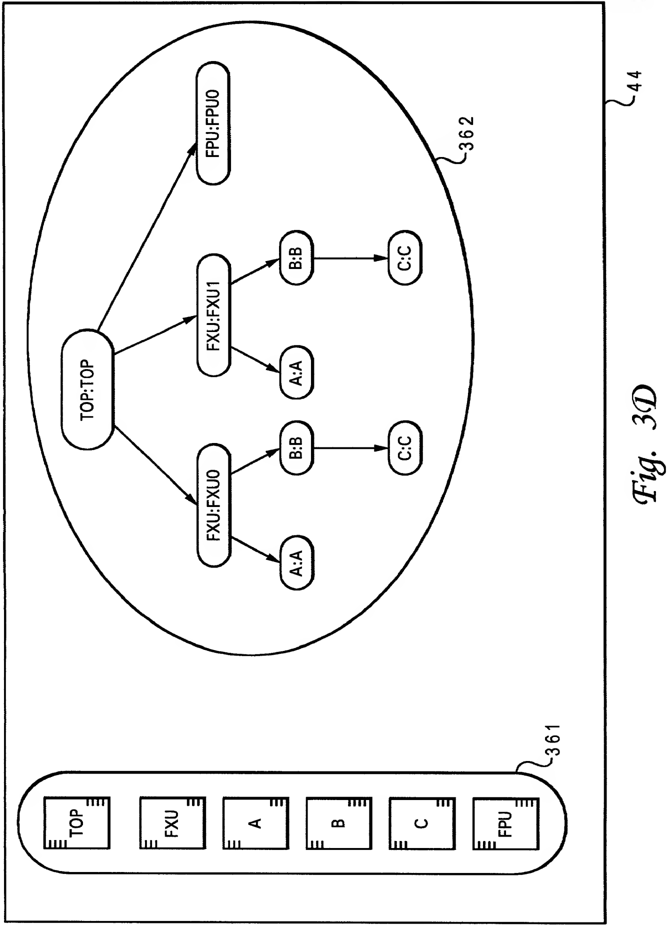

TOP:TOP
FXU:FXU0
FXU:FXU1
FPU:FPU0
A:A
B:B
C:C
A:A
B:B
C:C
TOP
FXU
A
B
C
FPU
361
362
44

Fig. 3D

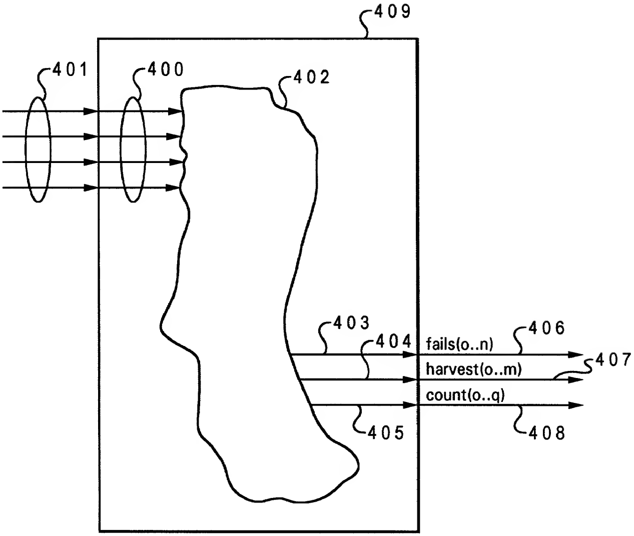

409
401
400
402
403
404
fails(o..n)
406
harvest(o..m)
407
count(o..q)
405
408

Fig. 4A

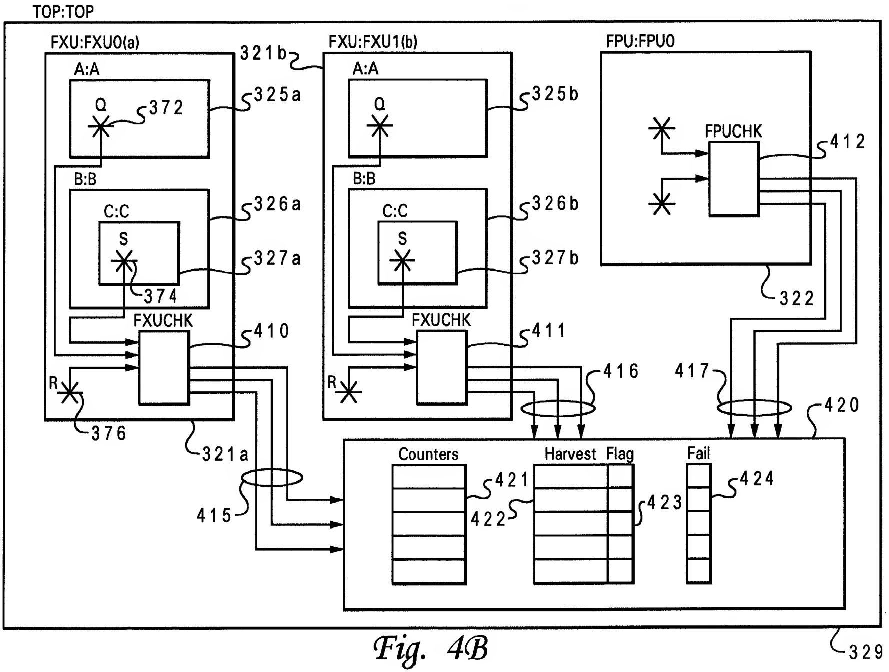
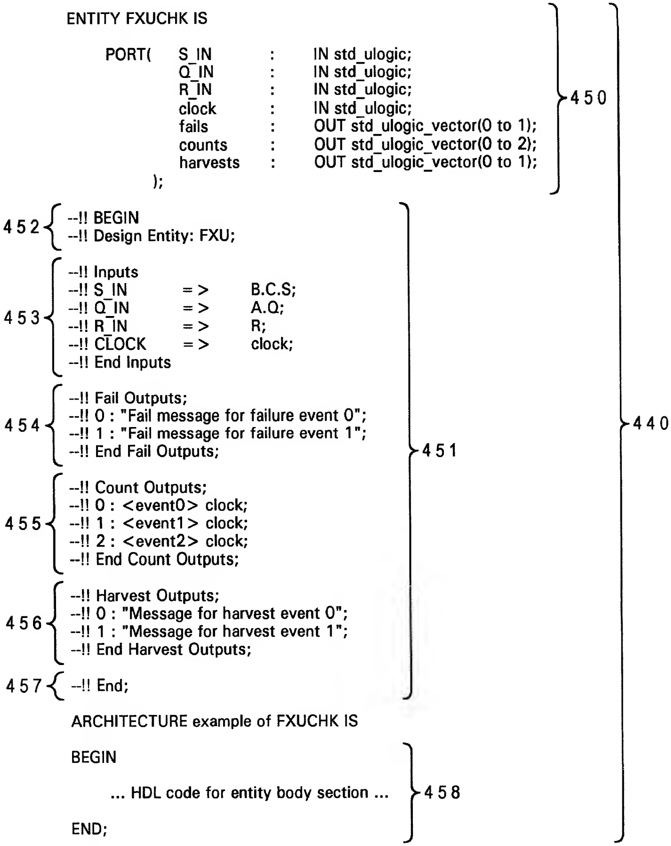

TOP:TOP
FXU:FXU0(a)
321b
FXU:FXU1(b)
FPU:FPU0
A:A
Q
372
325a
B:B
C:C
S
326a
327a
374
FXUCHK
410
R
376
321a
415
A:A
Q
325b
B:B
C:C
S
326b
327b
FXUCHK
411
R
416
FPUCHK
412
322
417
420
Counters
421
422
Harvest
Flag
423
Fail
424
329

Fig. 4B



```
ENTITY FXUCHK IS

      PORT(   S_IN        :      IN std_ulogic;
              Q_IN        :      IN std_ulogic;
              R_IN        :      IN std_ulogic;
              clock       :      IN std_ulogic;
              fails       :      OUT std_ulogic_vector(0 to 1);
              counts      :      OUT std_ulogic_vector(0 to 2);
              harvests    :      OUT std_ulogic_vector(0 to 1);
      );

--!! BEGIN
--!! Design Entity: FXU;

--!! Inputs
--!! S_IN        =>      B.C.S;
--!! Q_IN        =>      A.Q;
--!! R_IN        =>      R;
--!! CLOCK       =>      clock;
--!! End Inputs

--!! Fail Outputs;
--!! 0 : "Fail message for failure event 0";
--!! 1 : "Fail message for failure event 1";
--!! End Fail Outputs;

--!! Count Outputs;
--!! 0 : <event0> clock;
--!! 1 : <event1> clock;
--!! 2 : <event2> clock;
--!! End Count Outputs;

--!! Harvest Outputs;
--!! 0 : "Message for harvest event 0";
--!! 1 : "Message for harvest event 1";
--!! End Harvest Outputs;

--!! End;

ARCHITECTURE example of FXUCHK IS

BEGIN

      ... HDL code for entity body section ...

END;
```

Fig. 4C

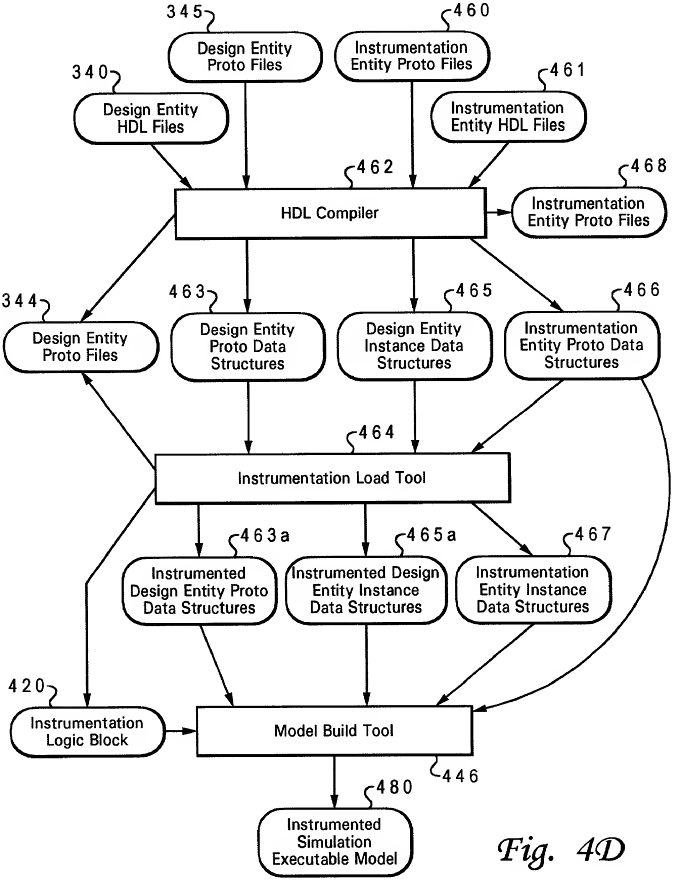

345
Design Entity Proto Files
460
Instrumentation Entity Proto Files
340
Design Entity HDL Files
461
Instrumentation Entity HDL Files
462
468
HDL Compiler
Instrumentation Entity Proto Files
344
Design Entity Proto Files
463
Design Entity Proto Data Structures
465
Design Entity Instance Data Structures
466
Instrumentation Entity Proto Data Structures
464
Instrumentation Load Tool
463a
Instrumented Design Entity Proto Data Structures
465a
Instrumented Design Entity Instance Data Structures
467
Instrumentation Entity Instance Data Structures
420
Instrumentation Logic Block
Model Build Tool
446
480
Instrumented Simulation Executable Model

Fig. 4D

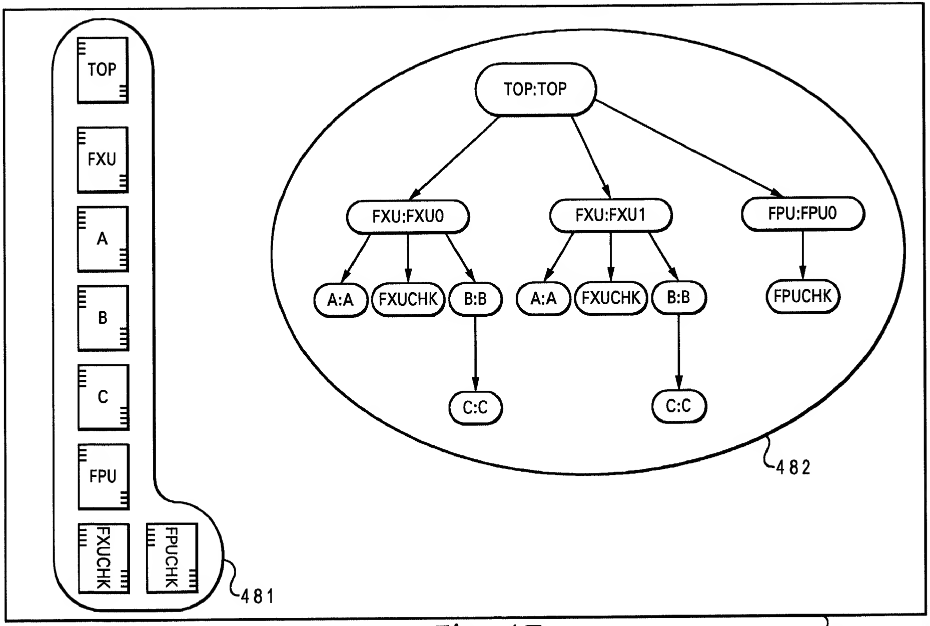

TOP
FXU
A
B
C
FPU
FXUCHK
FPUCHK
481
TOP:TOP
FXU:FXU0
FXU:FXU1
FPU:FPU0
A:A
FXUCHK
B:B
A:A
FXUCHK
B:B
FPUCHK
C:C
C:C
482
44

Fig. 4E

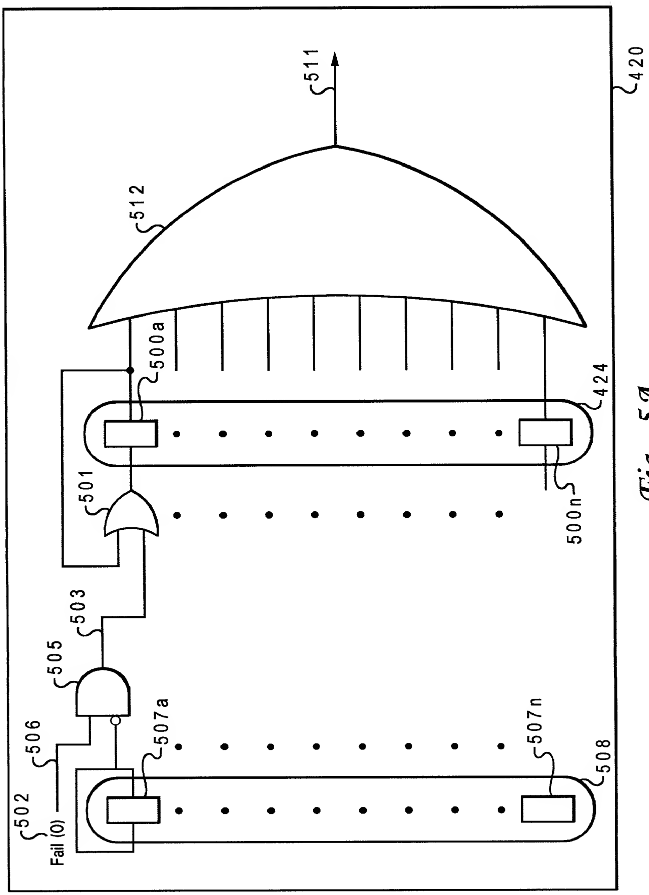

511
420
512
500a
424
501
500n
503
505
506
507a
507n
508
502
Fail (0)

Fig. 5A

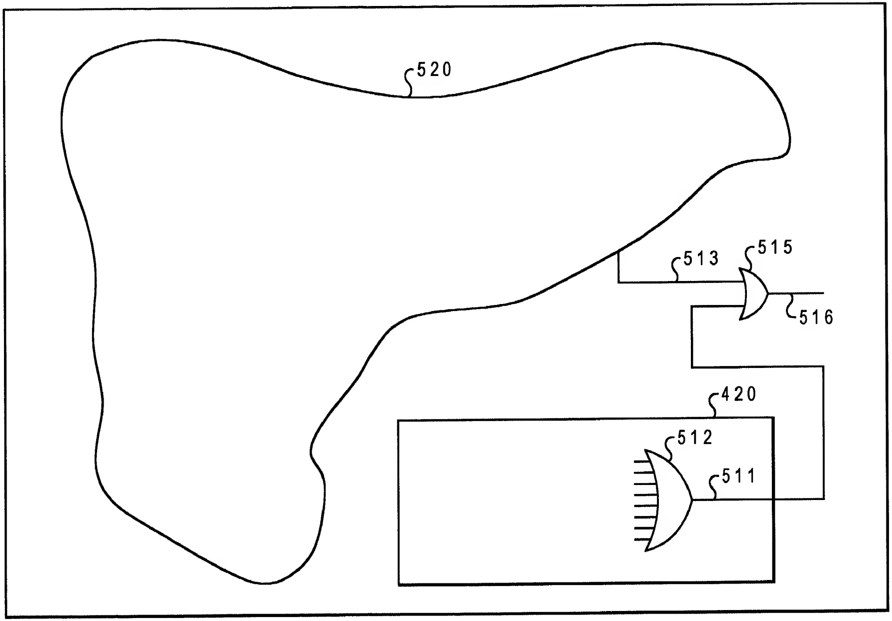


Fig. 5B

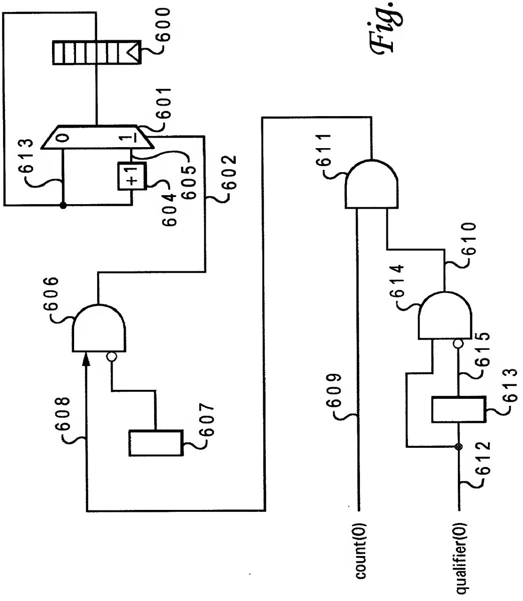

600
601
0
1
613
+1
605
604
602
606
607
608
611
610
614
615
613
612
609
count(0)
qualifier(0)

Fig. 6A

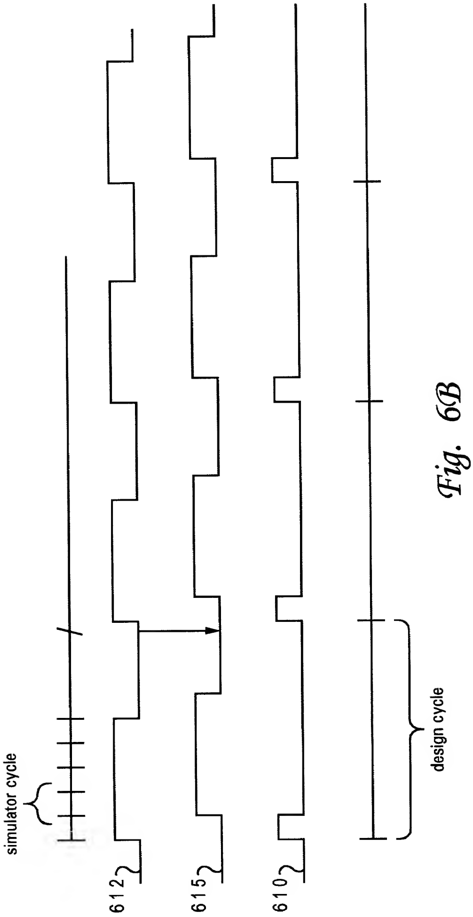


Fig. 6B

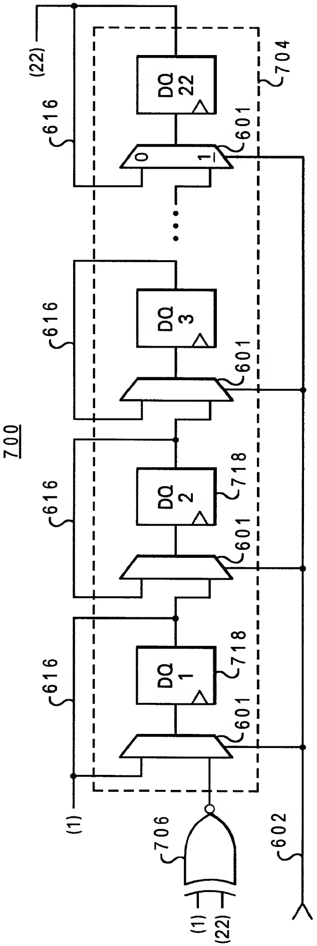

700
(22)
616
704
DQ 22
0
1
601
DQ 3
601
616
DQ 2
718
601
616
DQ 1
718
601
616
(1)
706
(1)
(22)
602

Fig. 7

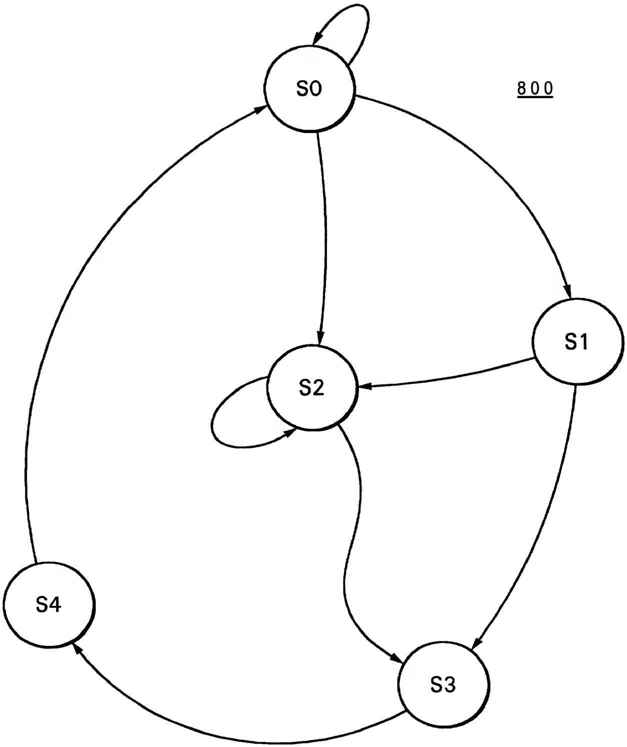


Fig. 8A
Prior Art

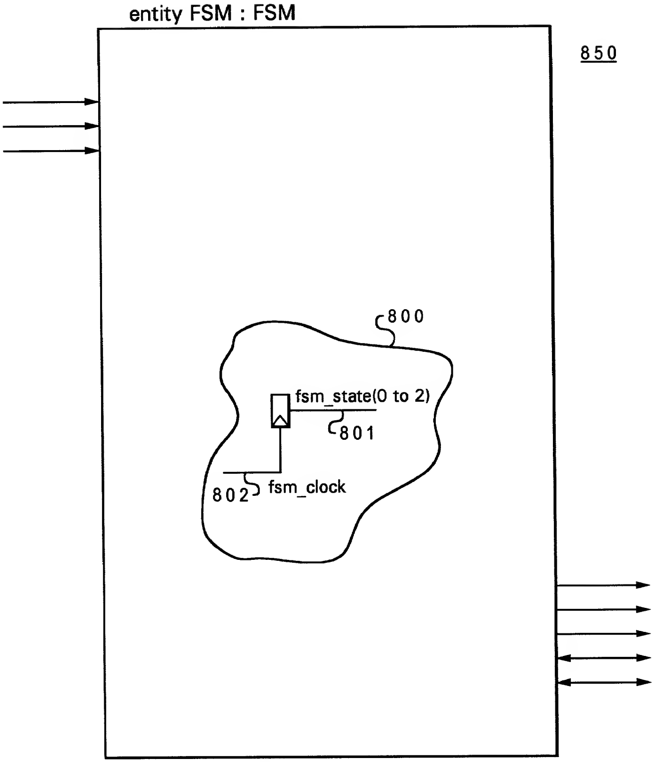
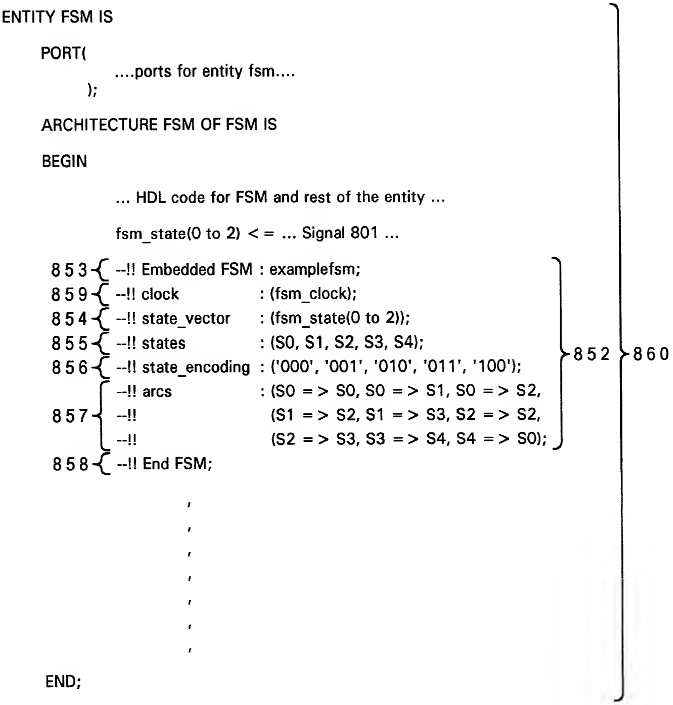


Fig. 8B
Prior Art



```
ENTITY FSM IS

      PORT(
                  ....ports for entity fsm....
             );

      ARCHITECTURE FSM OF FSM IS

      BEGIN

                  ... HDL code for FSM and rest of the entity ...

                  fsm_state(0 to 2) <= ... Signal 801 ...

853 { --!! Embedded FSM  : examplefsm;
859 { --!! clock           : (fsm_clock);
854 { --!! state_vector    : (fsm_state(0 to 2));
855 { --!! states          : (S0, S1, S2, S3, S4);
856 { --!! state_encoding  : ('000', '001', '010', '011', '100');
      --!! arcs            : (S0 => S0, S0 => S1, S0 => S2,
857 { --!!                   (S1 => S2, S1 => S3, S2 => S2,
      --!!                   (S2 => S3, S3 => S4, S4 => S0);
858 { --!! End FSM;

                         ,
                         ,
                         ,
                         ,
                         ,
                         ,
                         ,

      END;
```

Fig. 8C

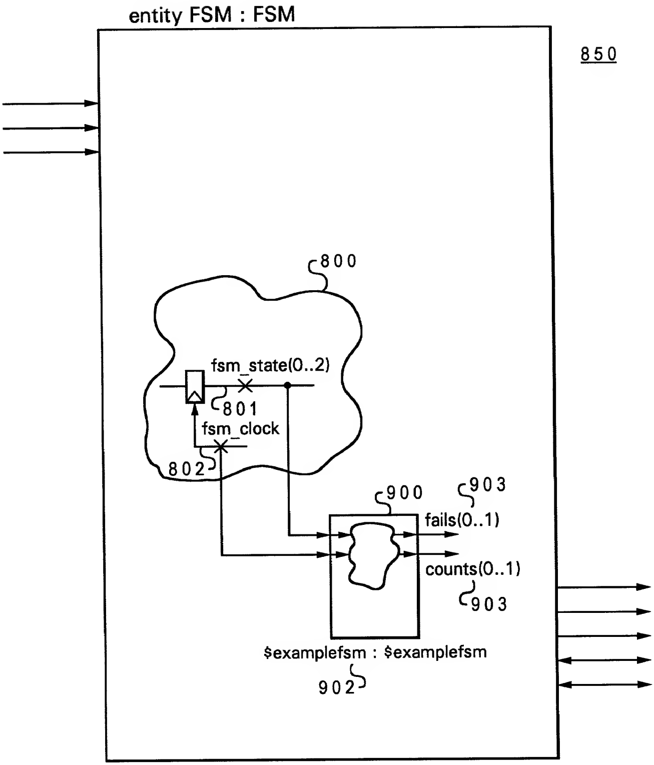
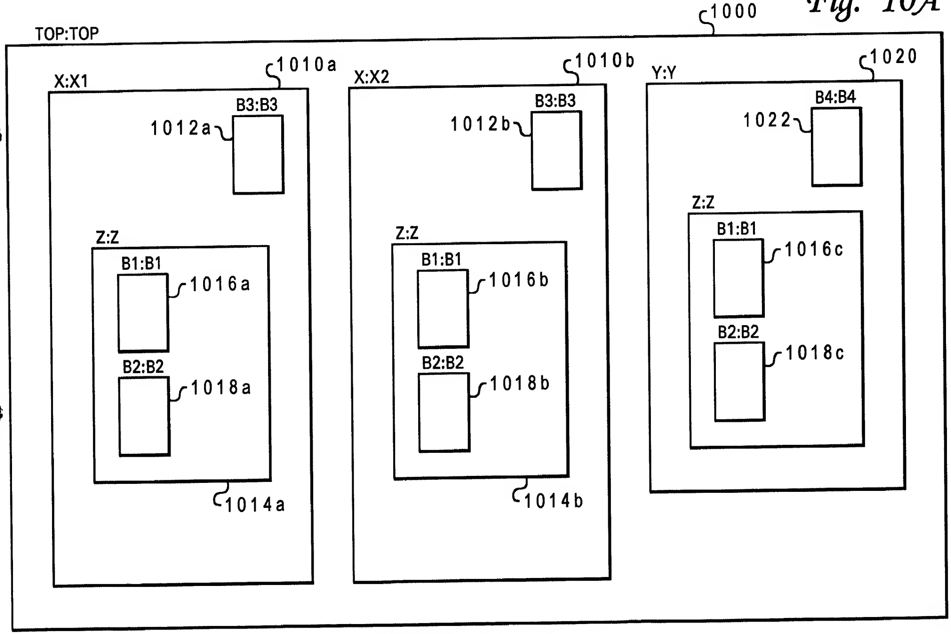

entity FSM : FSM
850
800
fsm_state(0..2)
801
fsm_clock
802
900
903
fails(0..1)
counts(0..1)
903
$examplefsm : $examplefsm
902

Fig. 9

Fig. 10A


1000
TOP:TOP
1010a
X:X1
B3:B3
1012a
Z:Z
B1:B1
1016a
B2:B2
1018a
1014a
1010b
X:X2
B3:B3
1012b
Z:Z
B1:B1
1016b
B2:B2
1018b
1014b
1020
Y:Y
B4:B4
1022
Z:Z
B1:B1
1016c
B2:B2
1018c

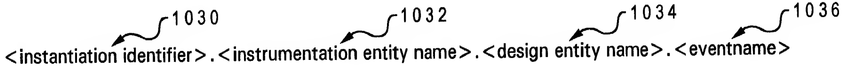
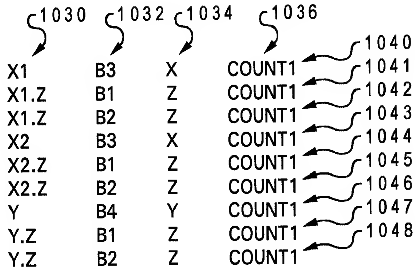

1030
1032
1034
1036
<instantiation identifier>.<instrumentation entity name>.<design entity name>.<eventname>

Fig. 10B



1030	1032	1034	1036
X1	B3	X	COUNT1
X1.Z	B1	Z	COUNT1
X1.Z	B2	Z	COUNT1
X2	B3	X	COUNT1
X2.Z	B1	Z	COUNT1
X2.Z	B2	Z	COUNT1
Y	B4	Y	COUNT1
Y.Z	B1	Z	COUNT1
Y.Z	B2	Z	COUNT1

Fig. 10C

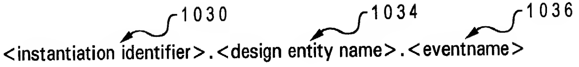


Fig. 10D

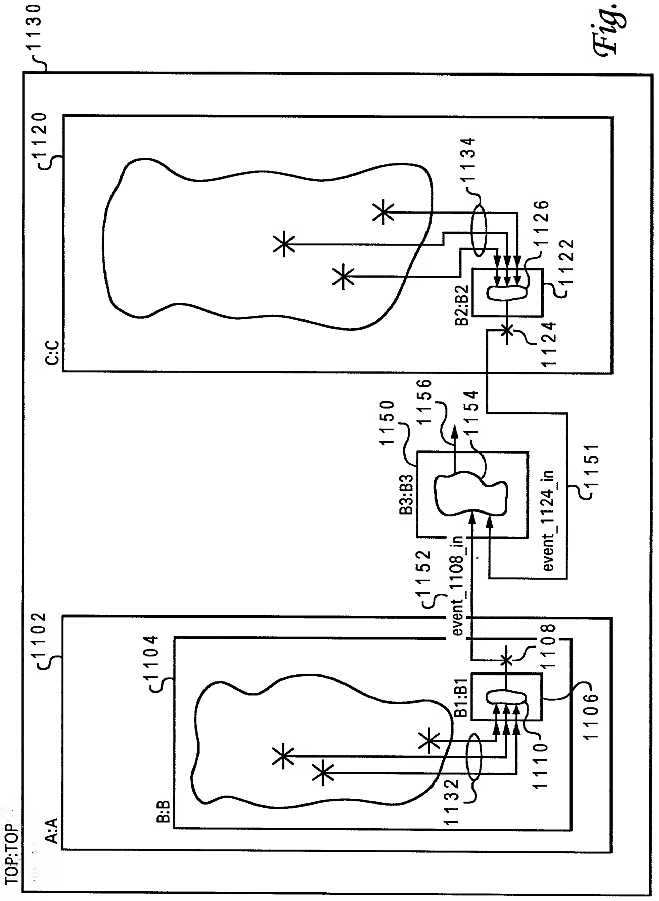
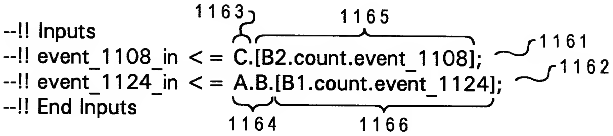
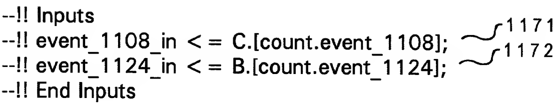

1130
1120
C:C
1134
1126
1122
B2:B2
1124
1150
1156
1154
B3:B3
1151
event_1124_in
1152
event_1108_in
1102
1104
1108
B1:B1
1106
1110
1132
B:B
A:A
TOP:TOP

Fig. 11A



```
--!! Inputs
--!! event_1108_in < = C.[B2.count.event_1108];
--!! event_1124_in < = A.B.[B1.count.event_1124];
--!! End Inputs
```

Fig. 11B



```
--!! Inputs
--!! event_1108_in < = C.[count.event_1108];
--!! event_1124_in < = B.[count.event_1124];
--!! End Inputs
```

Fig. 11C

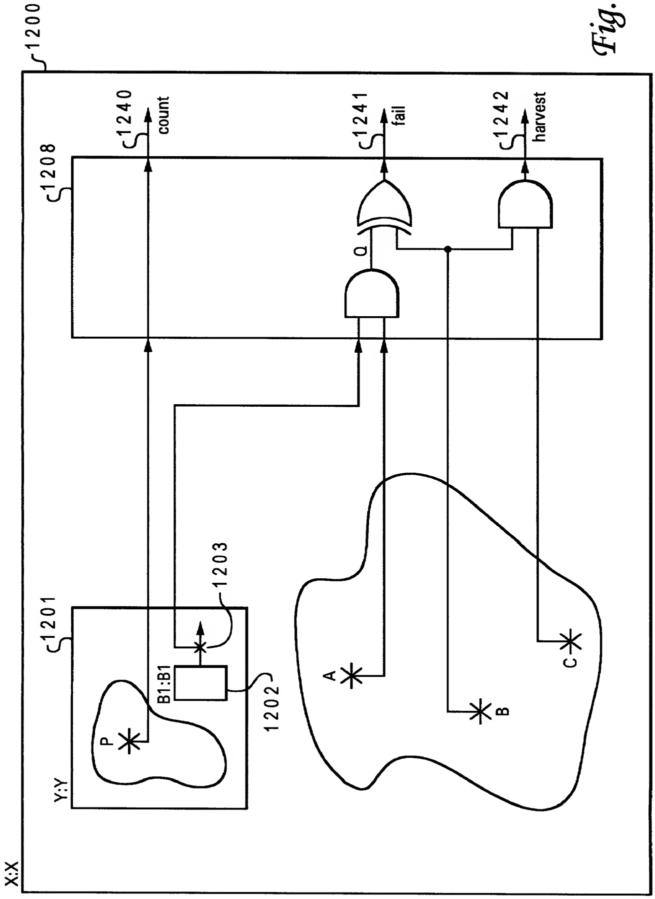
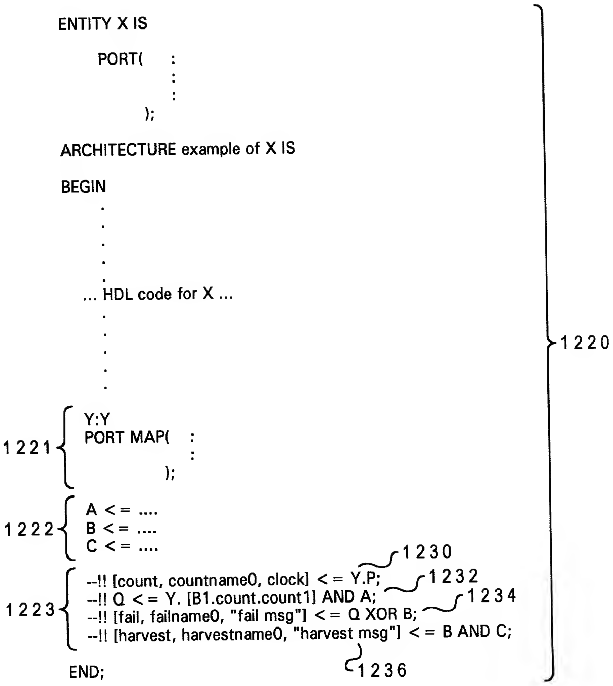

1200
1208
1240
count
1241
fail
1242
harvest
Q
1201
1203
B1:B1
1202
P
Y:Y
X:X
A
B
C

Fig. 12A



```
ENTITY X IS
        PORT(    :
                 :
                 :
              );
ARCHITECTURE example of X IS
BEGIN
      .
      .
      .
      .
      .
    ... HDL code for X ...
      .
      .
      .
      .
      .
     Y:Y
     PORT MAP(    :
                  :
               );
     A <= ....
     B <= ....
     C <= ....
     --!! [count, countname0, clock] <= Y.P;
     --!! Q <= Y. [B1.count.count1] AND A;
     --!! [fail, failname0, "fail msg"] <= Q XOR B;
     --!! [harvest, harvestname0, "harvest msg"] <= B AND C;
  END;
```

Fig. 12B

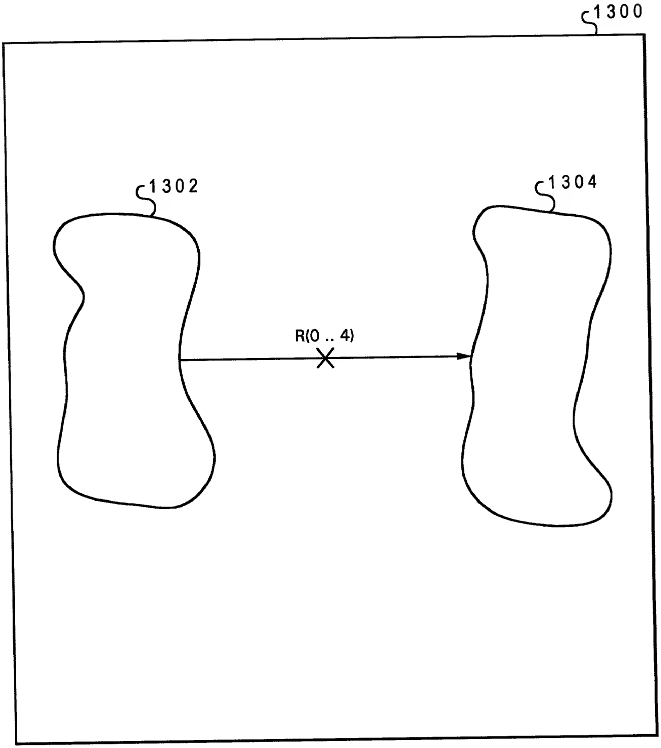


Fig. 13A

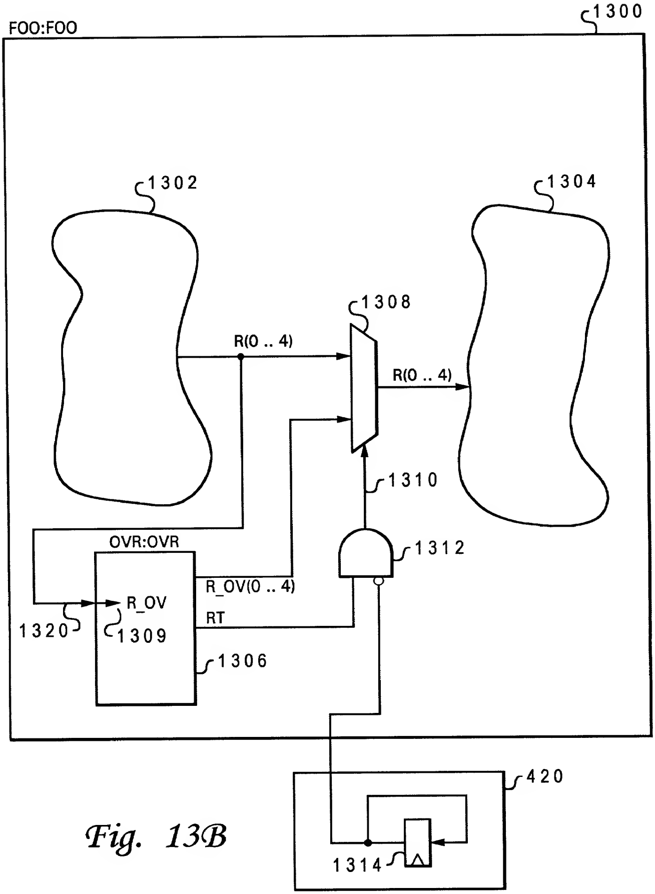
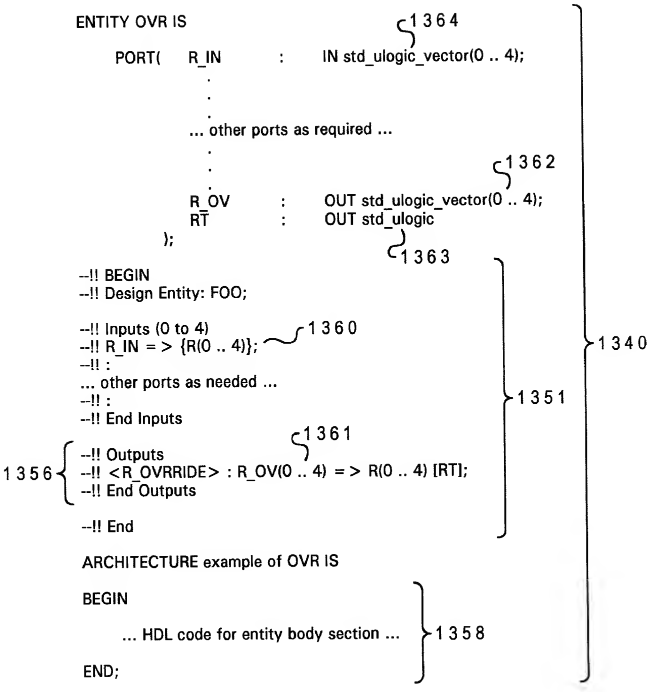
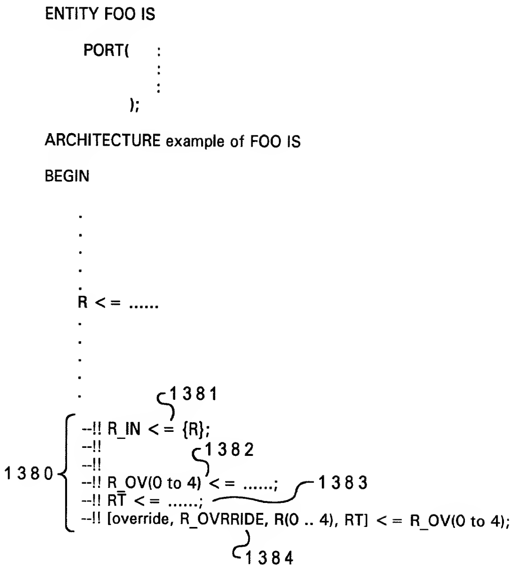

1300
FOO:FOO
1302
1304
1308
R(0 .. 4)
R(0 .. 4)
1310
1312
OVR:OVR
R_OV(0 .. 4)
RT
R_OV
1320
1309
1306
420
1314

Fig. 13B



```
ENTITY OVR IS

     PORT(    R_IN         :      IN std_ulogic_vector(0 .. 4);
                 .
                 .
                 .
              ... other ports as required ...
                 .
                 .
                 .
              R_OV         :      OUT std_ulogic_vector(0 .. 4);
              RT           :      OUT std_ulogic
          );

--!! BEGIN
--!! Design Entity: FOO;

--!! Inputs (0 to 4)
--!! R_IN = > {R(0 .. 4)};
--!! :
... other ports as needed ...
--!! :
--!! End Inputs

--!! Outputs
--!! <R_OVRRIDE> : R_OV(0 .. 4) = > R(0 .. 4) [RT];
--!! End Outputs

--!! End

ARCHITECTURE example of OVR IS

BEGIN

      ... HDL code for entity body section ...

END;
```

Fig. 13C



```
ENTITY FOO IS
        PORT(     :
                  :
                  :
            );
ARCHITECTURE example of FOO IS
BEGIN
        .
        .
        .
        .
        .
        R <= ......
        .
        .
        .
        .
        .
        --!! R_IN <= {R};
        --!!
        --!!
        --!! R_OV(0 to 4) <= ......;
        --!! RT̅ <= ......;
        --!! [override, R_OVRRIDE, R(0 .. 4), RT] <= R_OV(0 to 4);
```

Fig. 13D

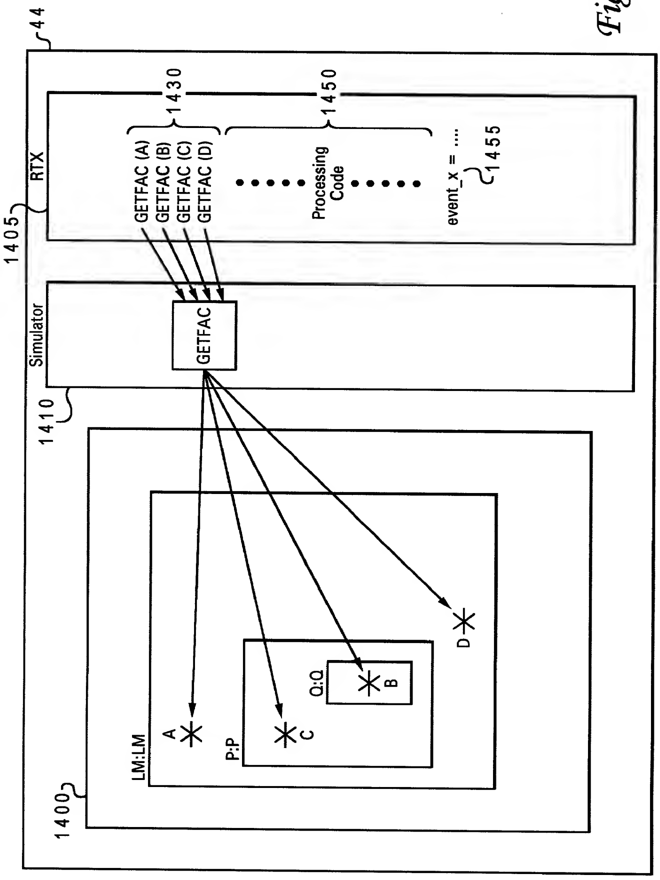

44
RTX
1405
1430
GETFAC (A)
GETFAC (B)
GETFAC (C)
GETFAC (D)
1450
Processing Code
event_x = ...
1455
Simulator
GETFAC
1410
1400
LM:LM
P:P
Q:Q
A
B
C
D

Fig. 14A

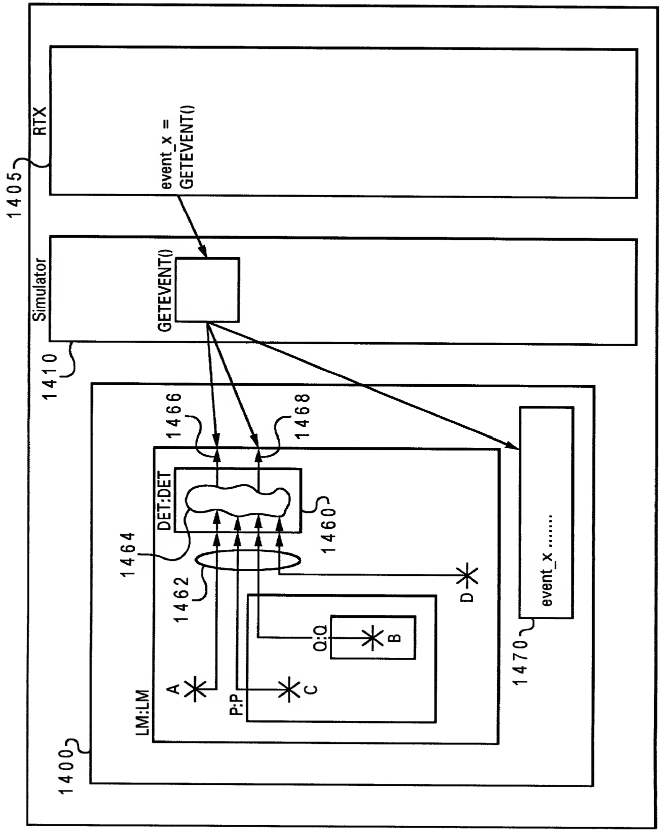
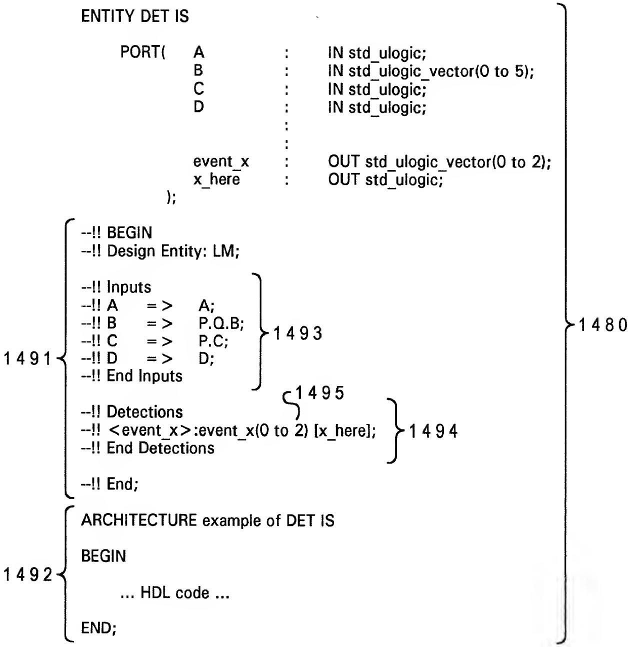

1405
RTX
event_x =
GETEVENT()
Simulator
1410
GETEVENT()
1400
1466
1468
DET:DET
1460
1464
1462
LM:LM
A
P:P
C
Q:Q
B
D
1470
event_x ....

Fig. 14B



```
ENTITY DET IS

        PORT(   A           :       IN std_ulogic;
                B           :       IN std_ulogic_vector(0 to 5);
                C           :       IN std_ulogic;
                D           :       IN std_ulogic;
                            :
                            :
                event_x     :       OUT std_ulogic_vector(0 to 2);
                x_here      :       OUT std_ulogic;
            );

--!! BEGIN
--!! Design Entity: LM;

--!! Inputs
--!! A      =>    A;
--!! B      =>    P.Q.B;
--!! C      =>    P.C;
--!! D      =>    D;
--!! End Inputs

--!! Detections
--!! <event_x>:event_x(0 to 2) [x_here];
--!! End Detections

--!! End;

ARCHITECTURE example of DET IS

BEGIN

        ... HDL code ...

END;
```

Fig. 14C